How to / Trackmania / Manialink / Building a Maniapage

Author : Marcel (Translations: JonTheKiller, Marcel)

LAST PAGE PAGE 1 2 3 4

Interesting aspects in the end

Known issues

Just after the release of TMF the first problems occur. To avoid similar questions I will list the known problems here.

*** Quality of images depends on texture settings**
Unfortunatly the images on the ManiaLink are displayed in dependency on the texture settings made in laucher. In plain language: Only players who set the texture quality on "high" see the ManiaLinks in high quality, all lower settings will lead to disfigurement of the images.

*** AddPlayerID can easily be manipulated**
As mentioned in the description of this attribute concerning quads and labels it can be manipulated in an easy way: by entering the URL manually. Critical areas must be secured in a different way.

*** Complete crash if the attribute folder is missing in the FileEntry**
Also this problem is named as important hint in the attribute description. If you leave out the attribute folder you will see your desktop immediately after you opened the ManiaLink. Avoid this bug by writing folder="" into the tag ;)

Combination of old and new ManiaLinks

As said in the beginning the old manialinks are compatible to TMF. This leads to new possibilities:

*** Complete new ManiaLink:**
Of course you can build your ManiaLink with these new elements I presented to you.

*** Complete old ManiaLink:**
Who bans the old system? If you do not need the new possibilities you can make your ManiaLink completely in the old system.
Secondary effect: The ManiaLink can also be viewed in TrackMania United!

*** Combinated ManiaLink:**
With the option of the "oldnew" (:D) ManiaLink you have the possibility to combine the easy table structure and the new elements.

This is easier then it sounds. Just split up your ManiaLink into two parts. Build the old ManiaLink as usual, remember only basic tags will work. Then you add the new elements in the end or in the beginning.

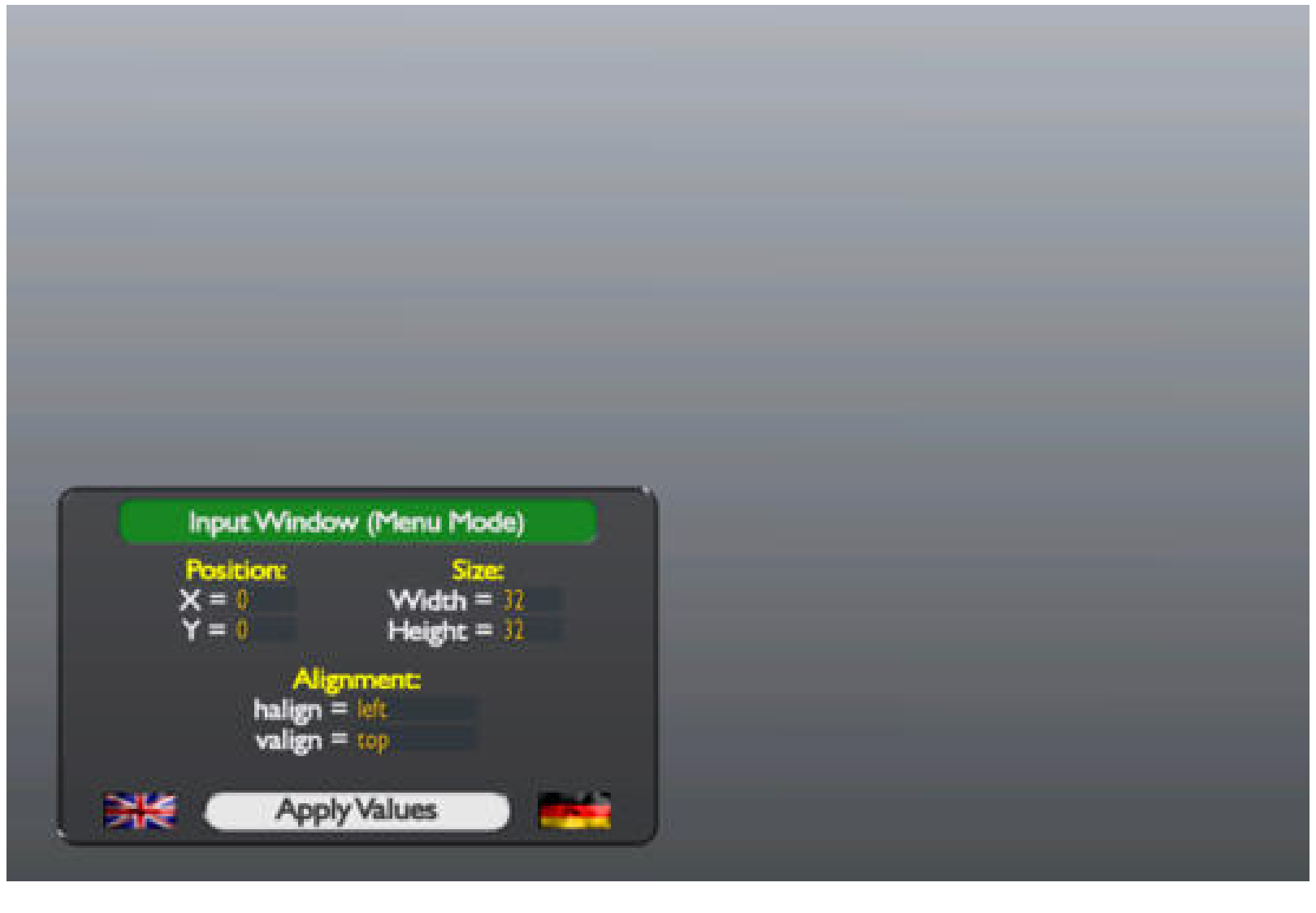
Thereby you can deal with the old ManiaLink part like a frame with the z-value: pos="X Y -0.04" or posn="X Y 2.56". in this way it is possible to place elements before or behind the old part.

Another hint: If you open a partially or complete new manialink in TMU you will see only the old part or nothing. There will be no error message in TMU.

Complex example

It's time to introduce a more extensive example of what we've learned so far. It's about the input windows on my quad-test-manialink. (**Manialink: Test Quad**).

An ingame screenshot of how it looks like:



To give a better overview, I shortened the Manialink-URLs, therefor not all of the input values are submitted anymore.

```
01 <?xml version="1.0" encoding="utf-8" ?>
02 <manialink>
03 <frame posn="-60 -10 5">
04 <label posn="25 -2 15" sizen="40 4" halign="center" style="TextTitle3" text="Input Window (Menu Mode)"/>
05 <quad posn="25 -1 10" sizen="40 4" halign="center" style="Bgs1InRace" substyle="BgTitle3_4"
manialink="http://funtrackers.bplaced.net/manialink/quad/index.php?type=classic&amp;lang=en&amp;x=inputx&amp;y=inputy"/>
06 <label posn="15 -6 5" halign="center" style="TextStaticSmall" text="$o$ff0Position:"/>
07 <label posn="15 -8.5 5" halign="right" style="TextStaticSmall" text="$oX = "/>
08 <label posn="15 -11 5" halign="right" style="TextStaticSmall" text="$oY = "/>
09 <entry posn="15 -8.5 5" sizen="5 2" style="TextValueSmall" name="inputx" default="0"/>
10 <entry posn="15 -11 5" sizen="5 2" style="TextValueSmall" name="inputy" default="0"/>
11 <label posn="35 -6 5" halign="center" style="TextStaticSmall" text="$o$ff0Size:"/>
12 <label posn="37 -8.5 5" halign="right" style="TextStaticSmall" text="$oWidth = "/>
13 <label posn="37 -11 5" halign="right" style="TextStaticSmall" text="$oHeight = "/>
14 <entry posn="37 -8.5 5" sizen="5 2" style="TextValueSmall" name="inputwidth" default="32"/>
15 <entry posn="37 -11 5" sizen="5 2" style="TextValueSmall" name="inputheight" default="32"/>
16 <label posn="25 -15 5" halign="center" style="TextStaticSmall" text="$o$ff0Alignment:"/>
17 <label posn="25 -17.5 5" halign="right" style="TextStaticSmall" text="$ohalign = "/>
18 <label posn="25 -20 5" halign="right" style="TextStaticSmall" text="$ovalign = "/>
19 <entry posn="25 -17.5 5" sizen="10 2" style="TextValueSmall" name="inputhalign" default="left"/>
20 <entry posn="25 -20 5" sizen="10 2" style="TextValueSmall" name="inputvalign" default="top"/>
21 <quad posn=" 7 -25.5 5" sizen="6 3" halign="center" image="http://funtrackers.bplaced.net/manialink/quad/en.png"
manialink="http://funtrackers.bplaced.net/manialink/quad/index.php?type=menu&amp;lang=en&amp;x=inputx&amp;y=inputy"/>
22 <quad posn="43 -25.5 5" sizen="6 3" halign="center" image="http://funtrackers.bplaced.net/manialink/quad/de.png"
manialink="http://funtrackers.bplaced.net/manialink/quad/index.php?type=menu&amp;lang=de&amp;x=inputx&amp;y=inputy"/>
23 <label posn="25 -25 5" halign="center" style="CardButtonMedium" manialink="http://funtrackers.bplaced.net/manialink/quad/index.php?
type=menu&amp;lang=en&amp;x=inputx&amp;y=inputy" text="Apply Values"/>
24 <quad sizen="50 30" style="Bgs1InRace" substyle="BgWindow2"/>
25 </frame>
26 </manialink>
```

Okay,let's go.

*** Lines 03 and 25 - The outline frame**
These lines create an outline frame, containing the entire input window. The purpose of this frame is, that I can relocate all elements within this frame to my needs, just by changing the frame's attributes. Doing so, all elements within the frame will move accordingly, because they're positioned relative to the frame's own position.
The frames' negative position values will place it at the left bottom of the screen.

*** Line 24 - The window itself**
The "quad" tag in Line 24 creates a little window, the one with the rounded egdes. It serves the sole purpose of visually joining all input fields and labels together.
Because it has no position values of its own, it will automatically inherit the outline frame's values. The "sizen" value determines the final size of the little window. The quad uses the Bgs1InRace style and BgWindow2 substyle.

*** Lines 04 and 05 - The title bar**
The "label" and the "quad" tags are responsible for shiny green title bar within the window. Both elements overlay each other, centered precisely, and the quad's lower z-value will make it appear behind the label.
The label utilises the "TextTitle3" style, showing the "Input Window (Menu Mode)" writing in bigger letters.
The quad utilises the "Bgs1Inrace" style with the "BgTitle3_4" substyle, which is green coloured, in both TMNF and TMUF.
In addition, the quad tag contains a manialink, pointing to the same .php file, but with new values (see line 23 for details)

*** Lines 06 to 08, 10 to 13 and 16 to 18**
These 9 labels are responsible for the description tags. Positioning so many labels properly can consume quite a lot of time, with lots of try'n'error involved.
Each word has its own "label" tag, using the "TextStaticSmall" style. The headlines use an additional "$o" formatting tag to make them bold, and the "$ff0" tag to make them appear in yellow color.

*** Lines 09, 10, 14, 15, 19 and 20**
Each input value has its own input field, using "input" + variable as the name to describe the functionality of the input field. The values for the variables will be sent to the .php file in line 23.
The "TextStaticSmall" style is responsible for the yellow and shadowed text in the input fields.
While X, Y, Width and Height are numbered values, Halign and Valign will only accept the text values "left", "center" and "right", respectively "top", "center" and "bottom". (This limitations have to be checked in the PHP script.)

*** Line 23 - Button and submission of value entries**
Here you can see how the values are submitted to a script (in this case, it's the same script again). The values in the URL are automatically replaced with the values from the "name" input fields before.
Note: multiple values are submitted by adding a "&" between them, although this character must be encoded as "&amp" for beeing properly interpreted by the PHP script.
The line "index.php?type=menu&amp;lang=en&amp;x=inputx&amp;y=inputy" would call the php script like this (with the default values of x = y = 0): index.php?type=menu&lang=en&x=0&y=0
Changing the values in the input fields will change the values in the processed URL accordingly.
The "CardButtonMedium" style creates a huge white button (with a predefined size), containing the "Apply values" writing.

*** Lines 21 and 22 - Language choice and images**
These 2 quads will display 2 flags, making it possible to chose the language of the interface. It requires the 'image' attribute, which points to an URL where the image is hosted (24-bit .png images are supported).
I could've used an "imagefocus" attribute as well to create a "mouseover" effect, but I have decided not to.
The tag's Manialink attribute points to the same URL, but with the language value of your choice (lang=de/en).

Direct linking via TMTP protocol

With the Forever Addon you now have the possibility to link your ManiaLinks, so that one single click is enough to call it. The only thing you have to do is using the TMTP protocol for linking. If you have selected the option during installation, all TMTP links will be sent to TMF when clicking (maybe you have to confirm a message of the browser). If TMF is not still running, it will open itself with this link, and the transmitted content will be displayed.

The use of TMTP links is as simple as HTTP links.

You have the following possibilities to link your ManiaLinks:

*** Call your ManiaLink with its full URL**
To call a ManiaLink with its complete URL, you have only to replace the http:// with tmtp://. After that, just type the domain and the folder structure, as it is known from normal HTTP-links.
Example: `tmtp://funtrackers.bplaced.net/manialink/quad/index.php`

*** Call your ManiaLink with its registered code**
It is also possible to call a ManiaLink with its registered code. Because these codes are only supported in TMUF, this version of linking will not work in TMNF. To link a registered code, just type tmtp:///: before it. The three / are important, otherwise some broswers will add an ending / which makes the code invalid and will return in an error ingame. The third slash avoids this unwanted behaviour.
Example: [url]tmtp:///:Marcel[/url]

Pay Attention to special characters: Because they are encoded in URLs, linking with them will not work. For example a space will be changed to "%20", so that tmtp:///:Test Quad will be sent as tmtp:///:Test%20Quad to TMUF. Because this code does not exist, TMUF is not able to translate this TMTP-link correctly.
By the way: You can also use the TMTP protocol to join a server and other things. But this is not part of the Tutorial, so I left it away.

Multilangual ManiaLink

It is possible to let the game select the suitable language on its own depending on the language the player set. The procedure is really basic: You create one file in which the single words, word groups or sentences are in different languages, the game selects the suitable one and shows it (respectively english is chosen if the suitable language is missing). Okay, it is not that easy because you have to stick to a certain syntax. The following example shows you the buildup of a multilanguage ManiaLink:

ManiaLink file:

```
<?xml version="1.0" encoding="utf-8"?>
<manialink>
<timeout>0</timeout>
<include url="http://localhost/manialink/lang.xml" />
<label posn="-20 0 1" textid="example" />
<label posn="-20 -10 1" textid="nadeo" />
<quad sizen="10 5" imageid="img" />
</manialink>
```

Here you see the buildup of the multilanguage ManiaLink. Important aspects are the include which loads the language file as well as the labels and the quad. Instead of the attribute text TextID is used, the text is not directly given, only a key which will also be found later in the lang.xml, same in the <quad> with the imageid. Nothing special up to now.

Language file: lang.xml

```
<dico>
<language id="de">
<example>Beispiel</example>
<nadeo>Danke Nadeo für diese neuen ManiaLinks</nadeo>
<img>http://localhost/manialink/de.png</img>
</language>
<language id="en">
<example>Example</example>
<nadeo>Thanks Nadeo for these new ManiaLinks</nadeo>
<img>http://localhost/manialink/en.png</img>
</language>
</dico>
```

Here you see the language file for the languages German and English.

Each language is enclosed throug <language id="CODE"></language>, whereas CODE is the game specific language code. Later you will see a list of the supported languages. Within the language tags the IDs are used: You can choose the content of the opening < > and the closing </ > tag on your own. The value you choose there is also the ID you use in the corresponging tag to get the text or the image. The content of the ID-tags will be directly used as value for the attribute, which uses this ID. You can use as much IDs as you wish.

Some hints:

*** The choice of the ID**
An ID must start with a letter. Afterwards you can use also numbers and an underline. Generally you should avoid special characters (except _). All other would be against the XML-syntax.

*** Missing IDs in a language**
If an ID is not found in the language, the player uses, the game searches this ID in English language. If it is available there it will be used, otherwise nothing would be shown. The order is: Selected Language > English > Nothing. So make sure at least English is complete!

*** Special characters**
If you use special characters in the lang.xml (as normal text, not as ID!) you figure out they won't be displayed correctly. To avoid this you have to save the lang.xml UTF-8 encoded. This can be done in normal Saving-dialog.

Supported attributes
To use an ID, you have only to add "id" to the attribute. For example "text" simply will become "textid", "image" will be "imageid" and so on.

The following attributes supports multilanguage:
* **<quad>** - image, imagefocus, url, manialink
* **<label>** - text, url, manialink
* **<audio>** - data
* **<video>** - data

This list is based on own tests and is maybe not complete. If you find another supported attribute, just say it. Thanks to Konte, who gives an important hint for that ;)

List of language codes:
* cz - Czech
* de - German
* en - English
* es - Espanol
* fr - French
* hu - Hungarian
* it - Italian
* jp - Japanese
* kr - Croatian
* nl - Dutch
* pl - Polish
* pt - Portuguese
* ru - Russian
* sk - Slovak
* zh - Chinese
I have not tested all these codes, but for this languages there are language files for the game. Therefore I think they will be supported on ManiaLinks ;)

Using parameters in ManiaLinks

Together with kastun and Balmung I discovered the possibility, to give parameters to registerd ManiaLinks with which could be worked with a script. This very useful feature (in combination with PHP of course) should be explained in this part of the tutorial.

It is quite easy: PHP pros allready know it in this form: "http://www.example.com/page.php?id=15" The parameter 15 is submited in the variable "id" to page.php. This could be transferred rather 1:1 to ManiaLinks.

Register the page as ManiaLink, without any parameters:
Code: "Page" => URL: http://www.example.com/page.php
To acces the page with a paramter just visit Page?id=15. In this example you will get the value 15 with $_GET['id'].
That was all the magic: As on PHP files you can stick variables and values on ManiaLinks with ?. Remember more variables are added with &amp; (in TrackMania this will be a normal &).

All good things must come to an end...
Now we reached the end of my Tutorial. I have said everything I wanted to say, alluded to everithing I wanted to allude. Now it is your part to use this knowledge. Present your new, or old, ManiaLinks and show what you are able to do now :D

Of course I will be happy about any comment and critics to this tutorial (after I received many of them to the German version). If you have any questions, post them, I think I will not be the only one who want to answer them ;)

Finally I want to thanks those of you, who supported me while writing this tutorial:

While writing the German version, the Betatesters answered me some questions to better understand the new ManiaLinks. To improve the quality, the FT and the ECU clan supported me with pointing on erronous, ununderstandable or just bad written parts. Thank you for that :)

After completing the German tutorial, I decided to translate it to English to view the knowledge to more than only Germans. I want to thank all you helped me with the translation: FT»Dany, CyR4S and TStarGermany.
Special thanks goes to ECU kastun. He did not only helped a lot while writing the German version, he also translated nearly the half of the tutorial and helped checking the translated parts. Without his support, the tutorial would never be as big and detailled as it is now. Thank you :)

That's it, now it's your turn.
FT»Marcel

LAST PAGE PAGE 1 2 3 4

This site is friendly to stationary (PC/MAC) and mobile (Android/IOS) browsers
Trackmania, Shootmania, Questmania games & media © Nadeo & UbiSoft used under fair use policy
8,373,164 page views since January 2008. All tutorials, services, coding © Mania Creative 2008-2013